**NSURLSession下载和断点续传**

NSURLSession是iOS7之后新的网络接口，和经常用到NSURLConnection是类似的。在程序在前台时，NSURLSession与NSURLConnection可以相互的替代。但是当用户在对程序进行强制关闭的时候此时NSURLSession会默认的自动断开。相比而言NSURLSession的优势主要体现在后台操作时候，而且在最流行的框架AFNetworking中也对NSURLSession提供了更好的支持。

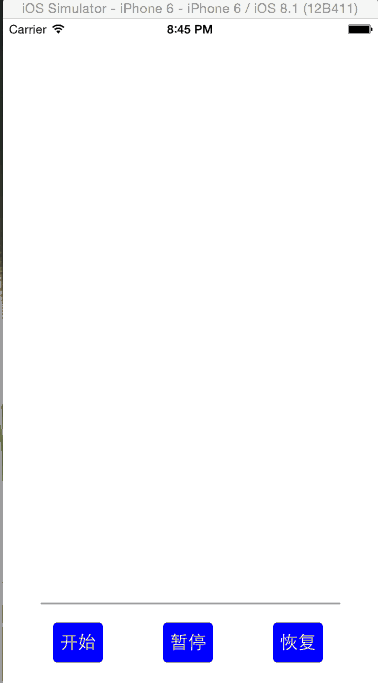
**主要提供的功能如下：**

1 下载文件到内存中

2 下载文件到路径

3 上传制定的文件等

**案例演示：图片下载断点续传**



**主要代码：**

**1、定义几个全局变量**

@interface ViewController ()

{

NSURLSessionDownloadTask \* \_task;

NSData \* \_data;

NSURLSession \* \_session;

NSURLRequest \* \_request;

UIProgressView \* \_pro;

UIImageView \* \_imageView;

}

**2、向视图中添加图片进度条**

\_imageView=[[UIImageView alloc] initWithFrame:CGRectMake(0, 0, 300, 300)];

\_imageView.center=self.view.center;

[self.view addSubview:\_imageView];

\_pro=[[UIProgressView alloc] initWithFrame:CGRectMake(\_imageView.frame.origin.x, \_imageView.frame.origin.y+400, 300, 40)];

**3、向视图中添加按钮（同样的方式添加三个）**

UIButton \* button=[[UIButton alloc] initWithFrame:CGRectMake(50, \_imageView.frame.origin.y+400+20, 50, 40)];

button.backgroundColor=[UIColor blueColor];

[button setTitle:@"开始" forState:UIControlStateNormal];

[button addTarget:self action:@selector(ddLoad) forControlEvents:UIControlEventTouchUpInside];

button.layer.borderWidth=1;

button.layer.borderColor=[UIColor blueColor].CGColor;

button.layer.cornerRadius=5;

[self.view addSubview:button];

**4、通过AFNetworkReachabilityManager网络状态监测**

- (void) \_checkNet{

//开启网络状态监控

[[AFNetworkReachabilityManager sharedManager] startMonitoring];

[[AFNetworkReachabilityManager sharedManager] setReachabilityStatusChangeBlock:^(AFNetworkReachabilityStatus status) {

if(status==AFNetworkReachabilityStatusReachableViaWiFi){

NSLog(@"当前是wifi");

}

if(status==AFNetworkReachabilityStatusReachableViaWWAN){

NSLog(@"当前是3G");

}

if(status==AFNetworkReachabilityStatusNotReachable){

NSLog(@"当前是没有网络");

}

if(status==AFNetworkReachabilityStatusUnknown){

NSLog(@"当前是未知网络");

}

}];

}

**5、开始下载**

- (void) ddLoad{

NSURLSessionConfiguration \* config=[NSURLSessionConfiguration defaultSessionConfiguration];

\_session=[NSURLSession sessionWithConfiguration:config delegate:self delegateQueue:nil];

//

NSURL \*url=[NSURL URLWithString:@src];

\_request=[NSURLRequest requestWithURL:url];

\_task= [\_session downloadTaskWithRequest:\_request];

NSLog(@"开始加载");

[\_task resume];

}

**6、设置暂停和回复**

- (void) pause{

//暂停

NSLog(@"暂停下载");

[\_task cancelByProducingResumeData:^(NSData \*resumeData) {

\_data=resumeData;

}];

\_task=nil;

}

- (void) resume{

//恢复

NSLog(@"恢复下载");

if(!\_data){

NSURL \*url=[NSURL URLWithString:@src];

\_request=[NSURLRequest requestWithURL:url]；

\_task=[\_session downloadTaskWithRequest:\_request];

}else{

\_task=[\_session downloadTaskWithResumeData:\_data];

}

[\_task resume];

}

**7、代理方法保存下载文件监控下载进度**

#pragma mark - delegate

- (void)URLSession:(NSURLSession \*)session downloadTask:(NSURLSessionDownloadTask \*)downloadTask didFinishDownloadingToURL:(NSURL \*)location{

NSURL \* url=[NSURL fileURLWithPath:@"/Users/jredu/Desktop/tt.png"];

NSFileManager \* manager=[NSFileManager defaultManager];

[manager moveItemAtURL:location toURL:url error:nil];

dispatch\_async(dispatch\_get\_main\_queue(), ^{

NSData \* data=[manager contentsAtPath:@"/Users/jredu/Desktop/tt.png"];

UIImage \* image=[[UIImage alloc ]initWithData:data];

\_imageView.image=image;

UIAlertView \* alert=[[UIAlertView alloc] initWithTitle:nil message:@"下载完成" delegate:self cancelButtonTitle:@"OK" otherButtonTitles: nil];

[alert show];

}) ;

}

- (void)URLSession:(NSURLSession \*)session downloadTask:(NSURLSessionDownloadTask \*)downloadTask

didWriteData:(int64\_t)bytesWritten

totalBytesWritten:(int64\_t)totalBytesWritten

totalBytesExpectedToWrite:(int64\_t)totalBytesExpectedToWrite{

CGFloat progress=(totalBytesWritten\*1.0)/totalBytesExpectedToWrite;

dispatch\_async(dispatch\_get\_main\_queue(), ^{

\_pro.progress=progress;

}) ;

}